**Assignment: 1**
**Due date: Friday - 20 March 2020 – 9:00 pm**
Course: Object Oriented Programming 2019/2020
Instructors: Dr. Sherin Moussa & Dr. Sally Saad

Ain Shams University
Faculty of Computer
& Information Sciences
2rd year

## General instructions:

1) Submit **only** running code.
2) Your program must contain three packages (one for each question).
3) Each package contains the files of the classes illustrated in each problem. Also, it should have an extra class that has a main function for that problem.
4) You must adhere to the name of classes, attributes and functions which are mentioned in the problems' definitions.
5) Compress the whole project folder as a .rar file.
6) Rename the .rar file after your department as follows:

| | |
|---|---|
| • .rar file for General Department | "*[G]_yourSectionNumber_yourID_yourName.rar*". |
| • .rar file for Software Department | "*[S]_yourSectionNumber_yourID_yourName.rar*". |
| • .rar file for BIO Department | "*[B]_yourSectionNumber_yourID_yourName.rar*". |
| • .rar file for Cyber Security Department | "*[C]_yourID_yourName.rar*". |
| • .rar file for MultiMedia and AI Departments | "*[M]_yourID_yourName.rar*". |

7) Submit **only** the .rar file.
8) If you need to re-submit your assignment. You must type the assignment version number, where the version number is 2, 3,4 etc.… as the following:

| | |
|---|---|
| • For General Department | "*[G]_yourSectionNumber_yourID_yourName_VersionNumber.rar*". |
| • For Software Department | "*[S]_yourSectionNumber_yourID_yourName_VersionNumber.rar*". |
| • For BIO Department | "*[B]_yourSectionNumber_yourID_yourName_VersionNumber.rar*". |
| • For Cyber Security Department | "*[C]_yourID_yourName_VersionNumber.rar*". |
| • For MultiMedia and AI Departments | "*[M]_yourID_yourName_VersionNumber.rar*". |

9) Please use the link below to submit your assignment:
https://docs.google.com/forms/d/e/1FAIpQLSfEPtqrCpzmrLU_g9dDCvu6XB-YxLj0Gdf-Z6juinnJYDIB2Q/viewform?usp=sf_link

As per rule 1, if your code had any errors, your assignment will not be considered. **So,** if you have any errors in your assignment, use Google to try to resolve your errors and if you didn't find the problem, please contact your TA (maximum by Wednesday "18 March 2020") to help you.

**As per rule 6, if you didn't rename your uploaded file as mentioned above, the automation process will ignore your assignment.**

Marks will be deducted for not following the rules **(-1/instruction**).

*Good luck*
*OOP team 2020*

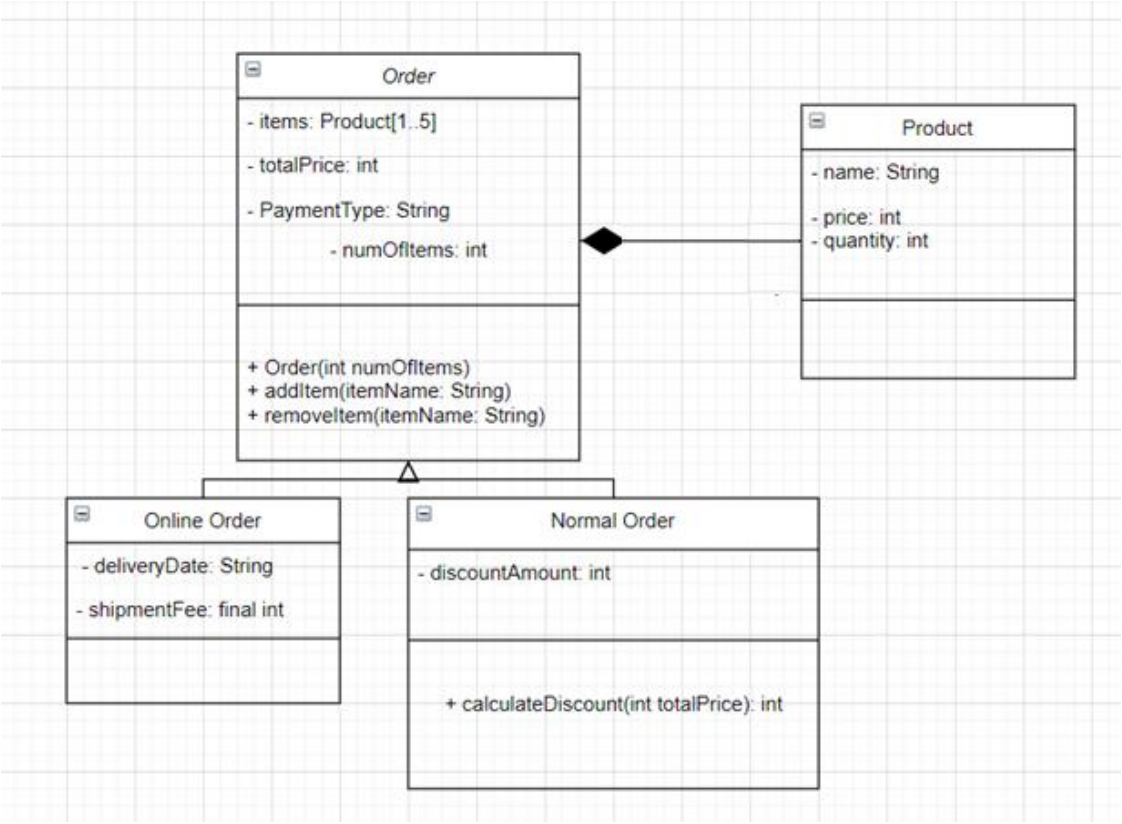
**Assignment: 1**
**Due date: Friday - 20 March 2020 – 9:00 pm**
Course: Object Oriented Programming 2019/2020
Instructors: Dr. Sherin Moussa & Dr. Sally Saad

Ain Shams University
Faculty of Computer
& Information Sciences
2$^{rd}$ year

## Assignment Questions

### Problem 1:

Implement the following UML class diagram.



Note the below points:

- Use Encapsulation (setters, getters) while implementing the classes.
- Calculatediscount() method should return the total price after removing 20% from its initial value.
- In the main function create two orders:
  - An Online Order that consists of 3 products of different prices and quantities.
  - A Normal Order that consists of 1 product.

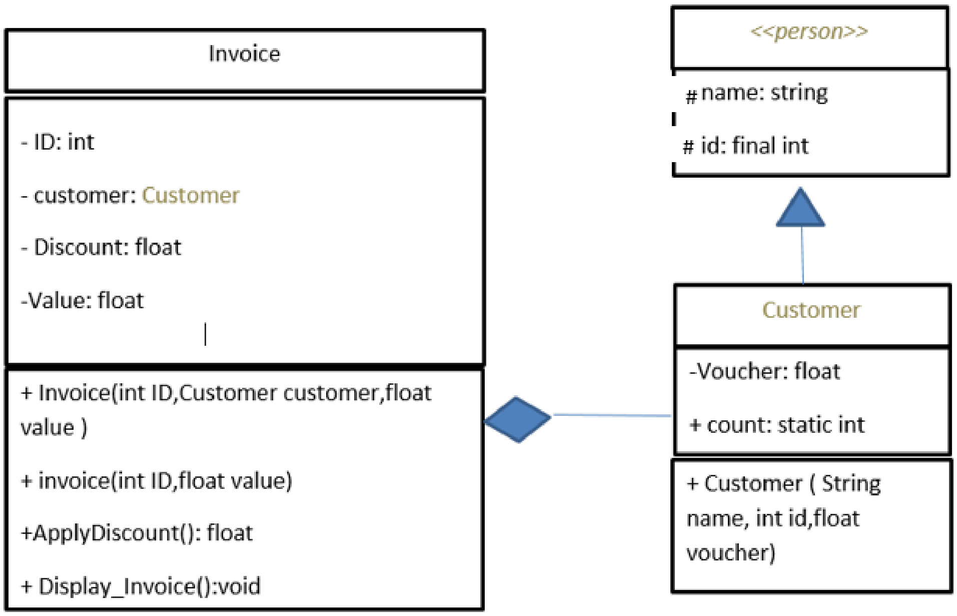
  And then display their details. After that, calculate the discount amount and display the total price after it.

- The class diagram may not be complete, so you are free to add any extra attributes and methods that are needed while implementation.

*Good luck*
*OOP team 2020*

## Problem 2:

Implement the following UML class diagram.



Note the below points:

- Use Encapsulation (setters, getters) while implementing the classes.
- Apply Chaining in the first constructor while implementing the invoice Class.
- Initialize the Discount in the first constructor with 0.05 , such that the function ApplyDiscount() :

    1. Will add 10% to the discount **for every 1000 L.E in the Value of the invoice** and apply discount to the Value.
    2. **If the Customer's voucher> 0, it will add the value of the Voucher to the Discount** and apply the discount to the value.
    3. return the Value

- Function Display_Invoice(): Prints all the data members of the invoice and the customers data.

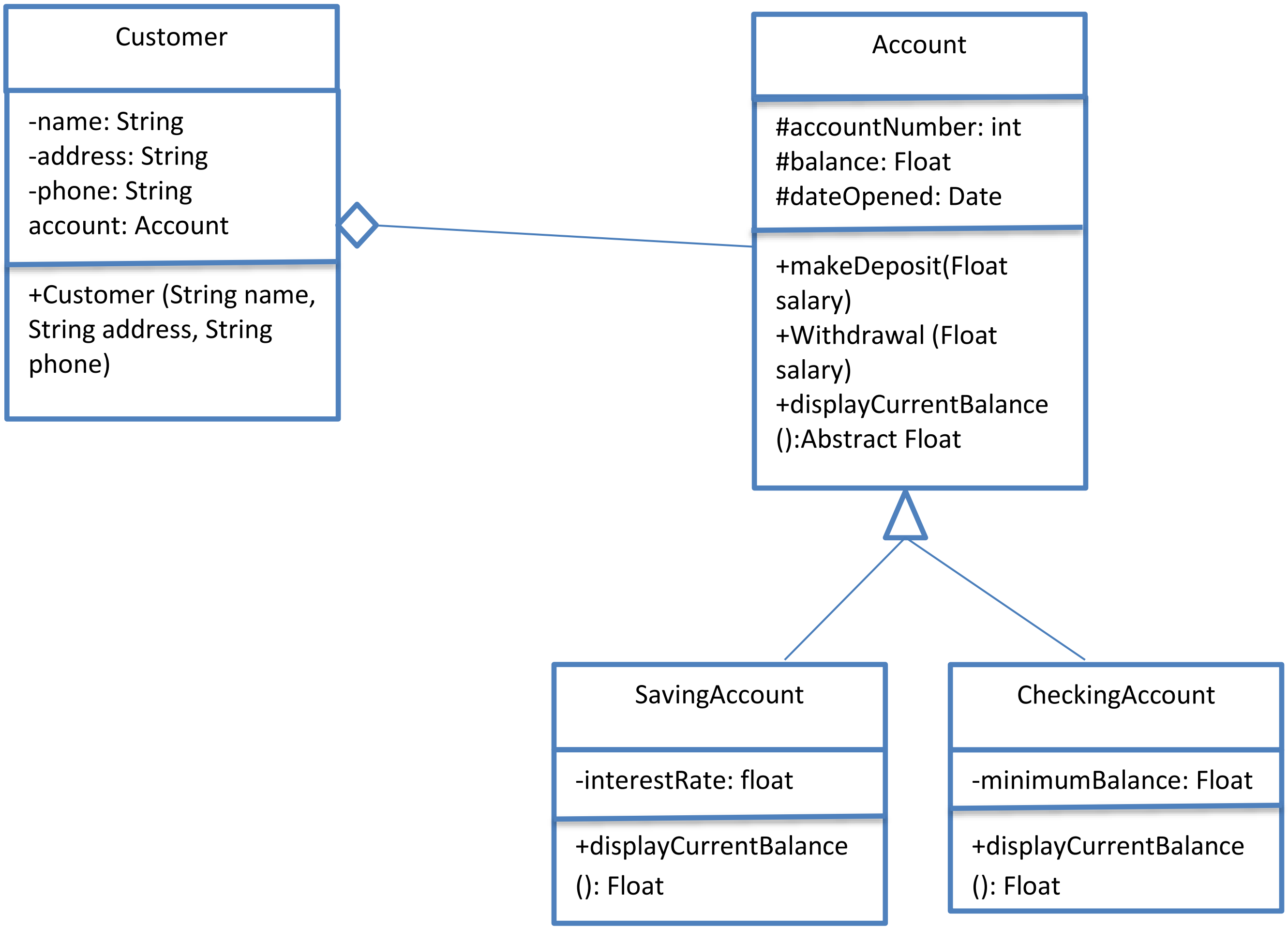
- In the public class that contains the main function do the following inside the main:

1. Create 2 Customers (customer 1: Ahmed, id:101,Voucher:0.0 – Customer 2: Ayman, id: 102. Voucher :0.02).
2. Display the count of created customers (should be equal to 2).
3. Create object 1 from class invoice (id:001, customer 1, 3500.0) and object 2 from class invoice (id:001,customer 2, 1000.0 using the first parametrized constructor.
4. Call ApplyDiscount and Display_Invoice for the two invoice objects.

*Good luck*
*OOP team 2020*

**Problem 3:**

Implement the following UML class diagram.


Customer
-name: String
-address: String
-phone: String
account: Account
+Customer (String name, String address, String phone)
Account
#accountNumber: int
#balance: Float
#dateOpened: Date
+makeDeposit(Float salary)
+Withdrawal (Float salary)
+displayCurrentBalance ():Abstract Float
SavingAccount
-interestRate: float
+displayCurrentBalance (): Float
CheckingAccount
-minimumBalance: Float
+displayCurrentBalance (): Float

- Use Encapsulation (setters, getters) while implementing the classes.
- Set interestRate of Saving Account with 0.7
- In the main function:
- Create two customers with two different accounts.
- And then display all the customers with all the information of their account.

Note that:

- you can't create CheckingAccount if the customer balance is less than minimum balance, so you have to check it first.
- **MakeDeposit**: method adds an amount on the current balance.
- **Withdrawal:** method draws a certain amount from the current balance.

*Good luck*
*OOP team 2020*

- **displayCurrentBalance**:in the SavingAccount class displays all the information of it along with the original balance and the balance value after adding the interest rate. (balance + balance*interestRate)
- **displayCurrentBalance** method displays all the information of the CheckingAccount.

**Also, you should apply the runtime polymorphism while solving this problem.**

*Good luck*
*OOP team 2020*